Supplementary Material of ReconfigISP: Reconfigurable Camera Image Processing Pipeline

Ke Yu[1,2] Zexian Li[1,3] Yue Peng[1] Chen Change Loy[4] Jinwei Gu[1,5]

[1]SenseTime Research and Tetras.AI [2]CUHK-SenseTime Joint Lab, The Chinese University of Hong Kong

[3]Beihang University [4]S-Lab, Nanyang Technological University [5]Shanghai AI Laboratory

ericyu16@hotmail.com lizexian0427@gmail.com

{pengyue, gujinwei}@tetras.ai ccloy@ntu.edu.sg

Abstract

In the supplementary material, we first specify more details of our experiments. Then, we demonstrate that ReconfigISP can adjust imaging effects by tuning the parameters of each module. Moreover, the analyses of online pruning are presented. Finally, we show additional experimental results.

1. Experimental Details

In this section, we specify ISP modules, the proxy network architecture and training details.

ISP Modules. We focus on common algorithms including denoising, demosaicing, tone-mapping and white balancing. Denoising algorithms are essential to address the noise introduced by CMOS sensors. BM3D [3] and Path-Restore [15] are representative of traditional and deep methods, respectively. Classic algorithms such as Bilateral [13], Median [8] and NLM [1] are also considered. Demosaicing algorithm converts raw bayer pattern into standard RGB (sRGB) format. We implement traditional algorithms [6, 14] and a deep DemosaicNet [7]. Gamma and Tone Mapping algorithms compress a wide-range image into limited range or highlight the dark/bright part of the image. Specifically, Reinhard [9] and Crysisengine adopt simple formulas while Filmic uses complicated operations to fit a saddle function [5]. White Balancing algorithms correct color bias caused by different sensor sensitivity and light condition. For example, Whitepatch [10] adjusts image color based on a reference white point.

Proxy Network Architecture. We use SRCNN [4], a widely used and lightweight model for super-resolution, as our proxy network. Specifically, SRCNN [4] has three convolutional layers, with kernel size 9, 5 and 5, respectively. The channels of two intermediate features are 64 and 32, respectively.

The number of input and output channels vary across image domains. There are four image channels (RGGB) in RAW domain. The input channel is 4+x and the output channel is 4, where x represents the number of input parameters for each module. For demosaicing, the input channel is 4 (RGGB) and the output channel is 12. A pixel shuffle layer [12] with up-sampling factor 2 is used to transfer 12 output channels to 3 image channels. As for the algorithms in sRGB domain, the input channel is 3+x and the output channel is 3, where x represents the number of module parameters. The input of global tone mapping and white balancing modules contain additional 9 channels, with the global minimum, mean and maximum for each channel. The proxy network architecture is summarized in Table 1.

Training Details. We first clarify the settings of image restoration. In the architecture search stage, we adopt a training patch size of 48×48 and a batch size of 16. In the parameter optimization stage, we use a larger patch size to fine-tune the algorithm parameters. Specifically, the patch size and batch size are 192×192 and 4, respectively.

Then we specify the settings of object detection. A larger patch size is adopted to include more objects. In the architecture search stage, we use a training patch size of 256×256 and a batch size of 4. In the parameter optimization stage, the patch size and batch size are 768×768 and 1, respectively.

2. Flexibility of ReconfigISP

Unlike existing fixed frameworks, we allow changes and rerouting in the ISP architecture. Further flexibility can be achieved by tuning the parameters of each module. Such adjustment is not available in deep networks that attempt to construct the ISP pipeline in a single model. In Fig. 1, we present an example of the SID dataset [2]. The searched ISP architecture is "Path-Restore-Bayer -> Laplacian -> Gamma -> WB-Quadratic -> WB-Linear". Given such a white-box pipeline, we can adjust the parameters to achieve

Table 1: Proxy network architecture.

Domain	Category	Algorithm	Proxy Architecture	Input Channel	Output Channel
RAW->RAW	Denoising	Bilateral-Bayer	SRCNN [4]	7	4
		Median-Bayer		5	
		NLM-Bayer		7	
RAW->sRGB	Demosaicing	Laplacian	SRCNN [4] + Pixel Shuffle	4	3
sRGB->sRGB	Denoising	Bilateral	SRCNN [4]	6	
		Median		4	
		NLM		6	
		BM3D		8	
	Global Tone Mapping	Reinhard		14	
		Crysisengine		13	
		Filmic		14	
	White Balance	Whitepatch		13	

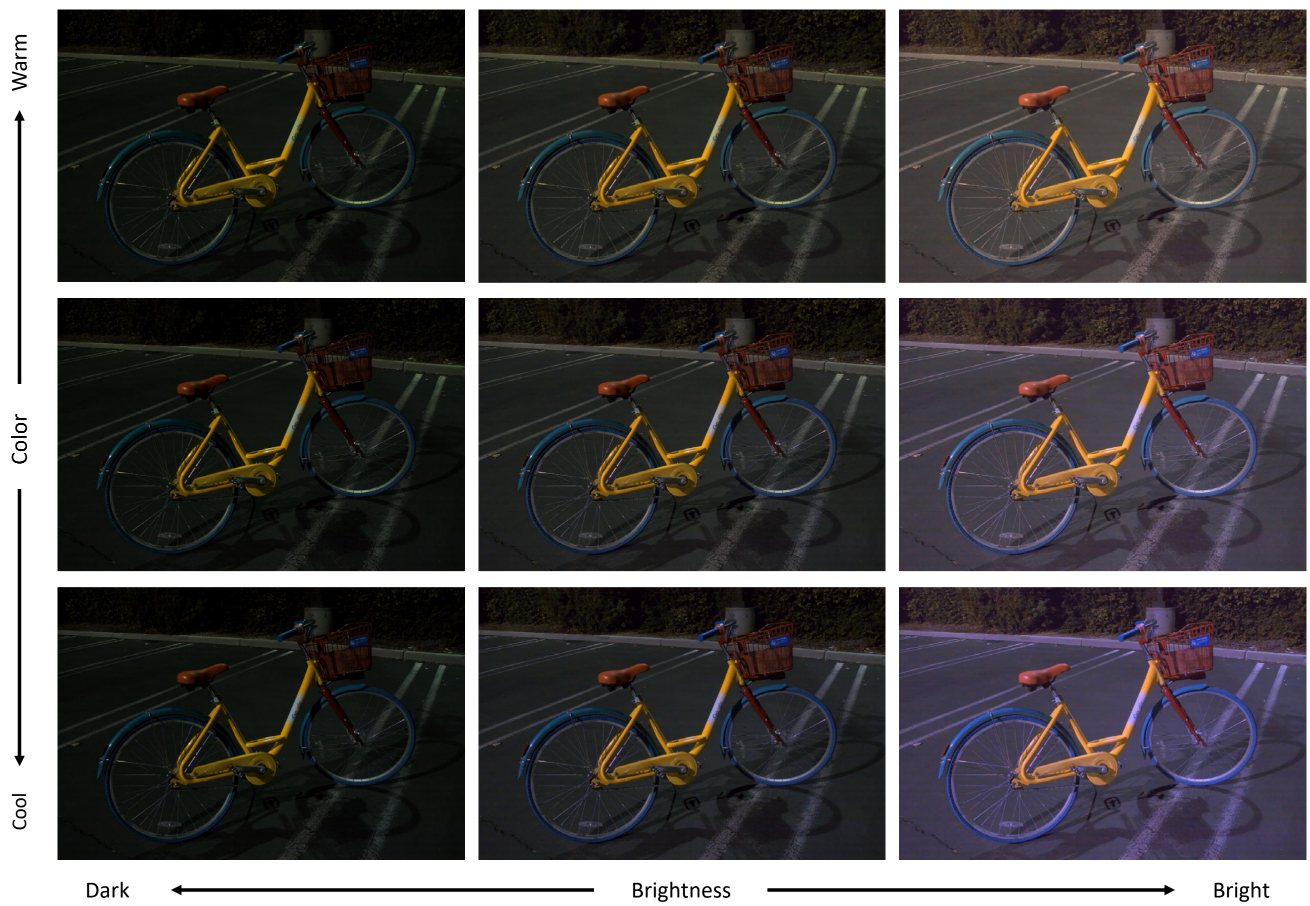

Figure 1: Thanks to the flexibility and modularity of ReconfigISP, one can perform post parameter tuning easily on each module (after the ISP architecture search) to adjust for the desired imaging effects. For example, we may tune the parameters of Gamma and WB-Linear to control brightness and color, respectively.

different imaging effects. For example, tuning the parameters of Gamma and WB-Linear can adjust brightness and color, respectively.

3. Analyses of Online Pruning

We present the training curve of architecture weights and illustrate the online pruning mechanism in Fig. 2. The curves represent the training procedure in the architecture

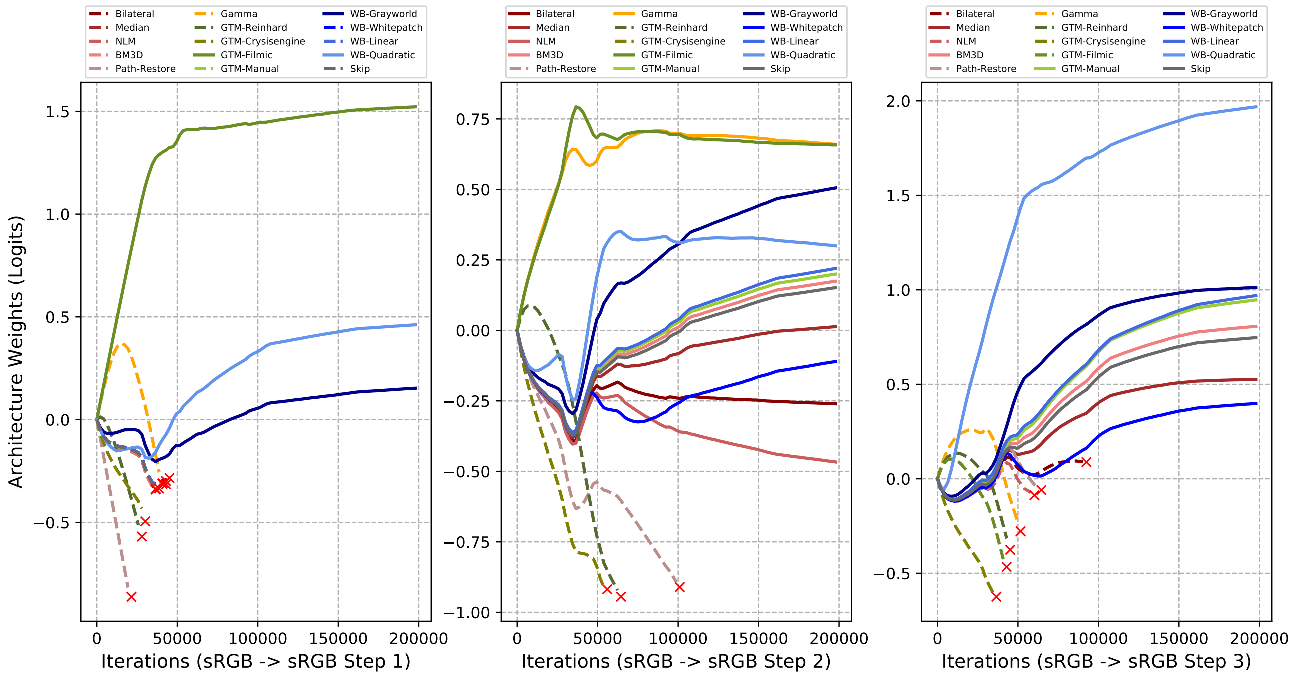

Figure 2: The training curve of architecture weights with online pruning. The experiment is conducted on S7 ISP dataset [11]. Three steps in sRGB domain are visualized. A dashed line represents a pruned algorithm, where it is cut off at the location denoted by a red cross (zoom in for best view).

Table 2: ISP architecture in different efficiency constraints. WB and GTM represent white balance and global tone mapping, respectively. The time stands for runtime on CPU to process one megapixel.

Dataset	Model Name	RAW->RAW	RAW->sRGB	sRGB->sRGB	Time (s)
SID [2]	ReconfigISP	Path-Restore-Bayer	Laplacian	Gamma, WB-Quadratic, WB-Linear	1.16
	ReconfigISP-Fast	-	Bilinear	Gamma, WB-Quadratic, NLM	0.63
	ReconfigISP-Faster	-	Bilinear	Gamma, WB-Linear	0.049
S7 ISP [11]	ReconfigISP	Path-Restore-Bayer	Nearest	GTM-Filmic, Gamma, WB-Quadratic	1.53
	ReconfigISP-Fast	-	Nearest	Gamma, WB-Quadratic, NLM	0.61
	ReconfigISP-Faster	-	Nearest	Gamma, WB-Linear	0.031

search stage on S7 ISP dataset [11]. We visualize the three steps in sRGB domain. Each curve denotes the architecture weight (before Softmax activation) of a specific algorithm. A dashed line terminated at a red cross indicates that the corresponding algorithm is pruned. In the first step, 12 out of 15 algorithms are pruned before 50,000 iterations. In the second step, the algorithms are competing fiercely, and only three algorithms are pruned. In the final step, seven algorithms are pruned ahead of the midpoint. Considering the complexity of each algorithm, about half of the computations are saved with the help of online pruning.

4. Additional Results

In Table 2, we show the searched ISP architectures under different efficiency constraints. ReconfigISP adopts a deep model Path-Restore-Bayer for denoising; ReconfigISP-Fast uses a more efficient NLM algorithm for denoising; ReconfigISP-Faster does not apply any denoising module. The efficiency of an ISP mostly depends on the complexity of denoising algorithms.

In Fig. 3 and 4, we further present qualitative results of image restoration and object detection, respectively. Our method yields cleaner restoration details and more accurate detection results compared to traditional ISP pipelines.

References

[1] Antoni Buades, Bartomeu Coll, and Jean-Michel Morel. A non-local algorithm for image denoising. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2005. 1

[2] Chen Chen, Qifeng Chen, Jia Xu, and Vladlen Koltun. Learning to see in the dark. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 1, 3, 5

[3] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen O. Egiazarian. Image denoising with block-matching and 3d filtering. In *Electronic Imaging*, 2006. 1

[4] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Image super-resolution using deep convolutional networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2016. 1, 2

[5] Gabriel Eilertsen, Rafal Mantiuk, and Jonas Unger. A comparative review of tone-mapping algorithms for high dynamic range video. *Computer Graphics Forum*, 2017. 1

[6] Pascal Getreuer. Linear methods for image interpolation. *Image Processing On Line*, 2011. 1

[7] Michaël Gharbi, Gaurav Chaurasia, Sylvain Paris, and Frédo Durand. Deep joint demosaicking and denoising. *ACM Transactions on Graphics*, 2016. 1

[8] Thomas S. Huang, George J. Yang, and Gregory Y. Tang. A fast two-dimensional median filtering algorithm. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 1979. 1

[9] Erik Reinhard, Michael Stark, Peter Shirley, and James Ferwerda. Photographic tone reproduction for digital images. In *Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques*, 2002. 1

[10] Alessandro Rizzi, Carlo Gatta, and Daniele Marini. Color correction between gray world and white patch. In *IS&T/SPIE Electronic Imaging*, 2002. 1

[11] Eli Schwartz, Raja Giryes, and Alex M Bronstein. Deepisp: Toward learning an end-to-end image processing pipeline. *IEEE Transactions on Image Processing*, 2018. 3

[12] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 1

[13] Carlo Tomasi and Roberto Manduchi. Bilateral filtering for gray and color images. *Proceedings of the IEEE International Conference on Computer Vision*, 1998. 1

[14] Yi-Ming Wu, Chiou-Shann Fuh, and Jui-Pin Hsu. Color interpolation for single ccd color camera. *IEEE Southwest Symposium on Image Analysis and Interpretation*, 2004. 1

[15] Ke Yu, Xintao Wang, Chao Dong, Xiaoou Tang, and Chen Change Loy. Path-restore: Learning network path selection for image restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 1

Figure 3: Additional qualitative results of image restoration on SID dataset [2].

Figure 4: Additional qualitative results of object detection on OnePlus dataset (zoom in for best view).